Crash Course: Microsoft Copilot in VS Code for Jupyter Notebooks

1. What is Copilot in VS Code?

Microsoft Copilot (previously GitHub Copilot) is an AI assistant integrated into VS Code that helps with:

- Code completions and auto-suggestions
- Generating functions or entire code cells from comments
- Explaining or debugging code
- Writing documentation, tests, or data visualizations

When combined with Jupyter Notebooks, Copilot becomes a powerful assistant for data analysis, teaching, and experimentation.

2. Setup and Installation

Step 1. Install VS Code

Download and install Visual Studio Code from: https://code.visualstudio.com/

Step 2. Install the Python and Jupyter extensions

In VS Code:

1. Go to **Extensions (Ctrl+Shift+X)**

2. Install:

   - **Python** (by Microsoft)
   - **Jupyter** (by Microsoft)

These extensions allow you to open and run `.ipynb` notebooks directly inside VS Code.

Step 3. Install GitHub Copilot

In VS Code:

1. Open the **Extensions** tab.
2. Search for **GitHub Copilot** and install it.
3. Optionally, install **GitHub Copilot Chat** for chat-based assistance.

Step 4. Sign in to GitHub

When you first activate Copilot:

1. You'll be asked to sign in to your GitHub account.
2. Approve the authorization in your browser.
3. Return to VS Code and confirm the connection.

You need a valid Copilot subscription or access through your organization (for example, through Microsoft educational programs).

Step 5. Enable Copilot for Notebooks

Once everything is installed:

1. Open a Jupyter notebook (`.ipynb`) in VS Code.
2. Click the **Copilot icon** in the bottom-right status bar.
3. Make sure "Enable Copilot in Notebooks" is turned on.

---

3. Using Copilot in Jupyter Notebooks

Inline Suggestions

Type a comment describing what you want, for example:

```
# plot the correlation matrix for this dataframe
```

After pressing **Enter**, Copilot will suggest:

```python
import seaborn as sns
import matplotlib.pyplot as plt

sns.heatmap(df.corr(), annot=True, cmap='coolwarm')
plt.show()
```

Press **Tab** to accept or **Esc** to dismiss.

Copilot Chat

You can open the Copilot Chat panel using **Ctrl+I** or by clicking the chat icon in the sidebar.

You can ask Copilot questions such as:

- "Explain this cell."
- "Optimize this code."
- "Why am I getting a KeyError?"
- "Generate code to visualize null values in this dataset."

You can also highlight a section of code, right-click, and choose: **Ask Copilot → Explain / Fix / Optimize Selection**

Generating Code from Prompts

Inside a code cell:

```
# generate python code to train a decision tree classifier on the iris dataset
```

Press **Enter**, and Copilot will produce the full implementation.

Data Science-Specific Examples

Task	Example Prompt
Data Cleaning	`# remove missing values and encode categorical columns`
Visualization	`# plot feature importance of the trained model`
Model Training	`# train a logistic regression and evaluate using accuracy and confusion matrix`
Exploratory Data Analysis	`# summarize numerical columns and plot histograms`

4. Controlling Copilot Behavior

Use the **Command Palette (Ctrl+Shift+P)** and search for:

- **Copilot: Enable/Disable**
- **Copilot: Open Chat**
- **Copilot: Explain This Code**

You can also control Copilot per language in: `File → Preferences → Settings → GitHub Copilot`

---

5. Troubleshooting

Issue	Solution
No suggestions appearing	Ensure you are logged in and the extension is enabled
Notebook kernel not found	Install Python and select the correct environment in VS Code
Chat not available	Install the **GitHub Copilot Chat** extension
High CPU usage	Disable inline suggestions temporarily

6. Example Workflow

Example of using Copilot during data analysis:

1. Type:

```
# load the Titanic dataset and preview it
```

Copilot writes code using pandas.

2. Then type:

```
# visualize survival rate by gender
```

Copilot writes the seaborn code.

3. Then type:

```
# train a random forest classifier and evaluate accuracy
```

Copilot writes the complete model training and evaluation code.

This workflow makes it ideal for interactive teaching or guided lab sessions.

---

7. Recommended Settings

In your `settings.json` file (open via **Ctrl+Shift+P → Preferences: Open Settings (JSON)**):

```
{
  "github.copilot.enable": {
    "*": true,
    "plaintext": false,
    "markdown": true,
    "notebook": true
  },
  "editor.inlineSuggest.enabled": true
}
```